CATÓLICA
ESCOLA SUPERIOR DE BIOTECNOLOGIA
PORTO

# NEW APPROACH OF SKIN CANCER DETECTION USING A DEEP LEARNING METHOD

by

Miguel Almeida e Silva

March 2025

CATÓLICA
ESCOLA SUPERIOR DE BIOTECNOLOGIA
PORTO

# NEW APPROACH OF SKIN CANCER DETECTION USING A DEEP LEARNING METHOD

Thesis presented to *Escola Superior de Biotecnologia* of the *Universidade Católica Portuguesa* to fulfill the requirements of Master of Science degree in Biomedical Engineering

by
Miguel Almeida e Silva

Supervisor : Onur Parlak

Co-Supervisor: Pedro Rodrigues

March 2025

## Acknowledgments

I would like to express my deepest gratitude to my supervisor, Onur Parlak for his support, insightful guidance, and invaluable expertise throughout the development of this thesis. His encouragement and constructive feedback were instrumental in shaping the direction and quality of my research. Also, I would like to mention Onur Parlak initiative in organizing group activities which made the group closer and enriched everyone's knowledge in the scientific field.

I am also thankful to my co-supervisor Pedro Rodrigues for keeping in touch with me from Portugal. His support and knowledge in Deep learning were essential to guide me through my thesis.

My sincere appreciation extends to my colleagues and friends in the Parlak Lab, whose collaborative spirit and stimulating discussions enriched my academic experience. Special thanks to them for their assistance with insightful recommendations and help for my project and keeping an incredible environment in the laboratory.

Finally, I would like to thank my family and friends for their encouragement. Their unwavering belief in my abilities has been a constant source of motivation.

## Abstract

Skin cancer occurs when normal skin cell growth becomes abnormal—often triggered by ultraviolet light exposure—and early detection is critical for effective treatment. This thesis investigates a novel diagnostic approach that targets specific amino acids associated with skin cancer cells. Focusing on asparagine, aspartic acid, and glutamine, it was employed three electrodes made of gold, platinum, and carbon to measure these biomarkers using the Square Wave Voltammetry method. The resulting signals were converted into spectrograms to serve as input for a deep learning model.

A total of 1,350 spectrogram samples (150 per amino acid per electrode) were used to train and evaluate the model. For the gold electrode, the model achieved an accuracy of 82% (F1 score: 81%, precision: 84%, recall: 82%, Cohen's kappa: 73%, ROC AUC: 93%), excelling in the identification of aspartic acid and glutamine while underperforming on asparagine. The platinum electrode attained an overall accuracy of 80% with perfect classification of aspartic acid, whereas the carbon electrode reached 78% accuracy, performing best for asparagine. In tests with amino acid mixtures, the gold electrode reliably detected all components, whereas the platinum and carbon electrodes showed selective misclassifications.

These findings highlight that electrode material significantly influences spectral pattern recognition and suggest that combining optimized electrode selection with deep learning can enhance early skin cancer diagnosis. Furthermore, the study demonstrates that increasing sample volume improves model accuracy, providing a promising foundation for future diagnostic tools.



**Key words:** Skin cancer, Deep learning, Amino acids, Square Wave Voltammetry, Electrodes

## Resumo

O cancro da pele ocorre quando o crescimento normal das células cutâneas se torna anormal — frequentemente desencadeado pela exposição à luz ultravioleta — e a deteção precoce é fundamental para um tratamento eficaz. Esta tese investiga uma abordagem diagnóstica inovadora que se centra em aminoácidos específicos associados às células cancerígenas da pele. Focando em asparagina, ácido aspártico e glutamina, foram usados três elétrodos, fabricados em ouro, platina e carbono, para medir estes biomarcadores através do método de voltametria de onda quadrada. Os sinais resultantes foram convertidos em espectrogramas, que serviram como entrada para um modelo de Deep learning.

Foram utilizados, no total, 1.350 espectrogramas (150 por aminoácido e por elétrodo) para treinar e avaliar o modelo. No elétrodo de ouro, o modelo atingiu uma precisão de 82% (pontuação F1: 81%, precisão: 84%, recall: 82%, kappa de Cohen: 73%, ROC AUC: 93%), destacando-se na identificação de ácido aspártico e glutamina, mas apresentando desempenho inferior na deteção de asparagina. O elétrodo de platina obteve uma precisão global de 80%, com classificação perfeita de ácido aspártico, enquanto o elétrodo de carbono alcançou 78% de precisão, apresentando melhor desempenho na identificação de asparagina. Em testes com misturas de aminoácidos, o elétrodo de ouro detetou fiavelmente todos os componentes, ao passo que os elétrodos de platina e carbono mostraram classificações seletivas.

Estes resultados evidenciam que o material do elétrodo influencia significativamente o reconhecimento dos padrões espectrais e sugerem que a combinação de uma seleção otimizada de elétrodos com Deep learning pode potenciar a deteção precoce do cancro da pele. Para além disso, o estudo demonstra que o aumento do volume de amostras melhora a precisão do modelo, constituindo uma base promissora para futuros instrumentos diagnósticos.



**Palavras chave:** Cancro da pele, Deep learning, Aminoácidos, Voltametria de onda quadrada, Elétrodos

# Table of contents

## List of abbreviations and glossary

**ADAM** – Adaptive Moment Estimation

**AdaGrad** – Adaptive Gradient Algorithm

**AUC** – Area Under the Curve

**BCC** – Basal Cell Carcinoma

**Cohen's kappa** – A statistical measure of inter-rater reliability

**ET** – Electronic Tongue

**F1 Score** – The harmonic mean of precision and recall

**FN** – False Negative

**FP** – False Positive

**LLM** – Large Language Model

**NASNet** – Neural Architecture Search Network

**PBS** – Phosphate-Buffered Saline

**PSD** – Power Spectral Density

**ReLU** – Rectified Linear Unit

**RMSProp** – Root Mean Square Propagation

**ROC** – Receiver Operating Characteristic

**SCC** – Squamous Cell Carcinoma

**STFT** – Short-Time Fourier Transform

**SWV** – Square Wave Voltammetry

**TCA** – Tricarboxylic Acid

**TP** – True Positive

**TN** – True Negative

**UV** – Ultraviolet

# 1 Introduction

## 1.1 Background Information

Early diagnosis of skin cancer remains a significant challenge due to the difficulty of detecting the disease in its initial stages. However, early detection is crucial, as it dramatically increases the effectiveness of treatment and improves patient outcomes.

This thesis explores the potential of a deep learning approach for recognizing amino acids associated with skin cancer cells. Amino acids play a vital role in the biochemical reactions that provide energy for cancer cell proliferation, and their distinct behaviours make them suitable biomarkers for this purpose.

To implement this approach, three different electrodes—made of materials with unique responses to each amino acid—are used to collect data. The data obtained is processed and input into a deep learning model for training and validation, enabling the model to make predictions based on the amino acid profiles.

The results suggest that this novel diagnostic method shows promise in identifying skin cancer cells. Moreover, with additional data in the future, the accuracy and reliability of the model are expected to improve significantly.

## 1.2 Different forms of skin cancer

Skin cancer occurs when skin cells grow abnormally due to damage, primarily from ultraviolet (UV) radiation resulting from sun exposure. The three most common types of skin cancer are basal cell carcinoma (BCC), squamous cell carcinoma (SCC), and melanoma, each with distinct characteristics.

- **Basal Cell Carcinoma (BCC)**: The most common type of skin cancer, BCC often appears as a flesh-colored, pearl-like bump or a pinkish patch. It typically develops after prolonged sun exposure and is frequently found on the head, neck, and arms. Although it grows slowly, early treatment is crucial as it can cause significant damage if left untreated.
- **Squamous Cell Carcinoma (SCC)**: The second most common type, SCC is characterized by red, firm bumps or scaly patches. It tends to develop in areas with

frequent sun exposure, such as the face, neck, and ears. SCC can also grow profoundly and spread if not treated early.

- **Melanoma**: While less common, melanoma is the deadliest form of skin cancer, responsible for the majority of skin cancer-related deaths. It can develop from an existing mole or become a new dark spot. Due to its rapid growth and high likelihood of metastasis, early detection is critical.

Both BCC and SCC are treatable but can lead to disfigurement if not addressed promptly. Melanoma's potential for rapid spread, underscores the importance of early diagnosis and intervention (American Academy of Dermatology, 2025).

## 1.3 Areas of the skin where skin cancer appear

Skin cancer predominantly develops on areas frequently exposed to the sun, such as the scalp, face, ears, neck, lips, chest, arms, hands, and legs. However, it can also manifest in less exposed areas, including the palm, soles, beneath the fingernails and toenails, and even the genital region.

The increasing incidence of skin cancer, coupled with its potential severity, particularly in the case of melanoma, underscores the importance of early detection. This thesis explores advanced diagnostic techniques and technologies that can facilitate the early identification of skin cancer, thereby improving treatment outcomes and reducing the associated morbidity and mortality.

## 1.4 Diagnosis of skin cancer

One pre-diagnosis approach is to examine the skin regularly to see if there are any changes in the shape of the moles or spots. Distinguishing between malignant and non-malignant skin conditions can be challenging, as visual similarities often make an accurate diagnosis difficult. To assist in this process, clinicians commonly use a dermatoscope, a magnification device that enhances the visibility of skin structures and patterns that might indicate malignancy (Cancer Research UK, n.d.).

If the clinical evaluation suggests the possibility of skin cancer, a biopsy is performed to confirm the diagnosis. This involves taking a small tissue sample from the affected area, typically under local anaesthesia to ensure patient comfort. The size and location of the lesion usually determine the biopsy method used. The tissue sample is then sent to a laboratory for

histopathological examination under a microscope, which is considered the gold standard for confirming the presence of cancer cells and differentiating them from benign conditions (Cancer Research UK, n.d.).

Different biopsies can be performed to diagnose skin abnormalities, each tailored to the specific characteristics of the lesion and clinical suspicion. The main types of biopsies include the following :

### 1.4.1 Incisional Biopsy

An incisional biopsy involves the removal of a small portion of the abnormal area using a surgical knife. The procedure removes the entire skin thickness, allowing for a comprehensive tissue analysis (Cancer Research UK, n.d.).

### 1.4.2 Excisional Biopsy

An excisional biopsy removes the abnormal area, including a margin of surrounding healthy tissue. This technique is often used when the lesion is small enough to be entirely excised (Cancer Research UK, n.d.).

### 1.4.3 Punch Biopsy

A punch biopsy is performed using a specialized instrument that extracts a small, circular skin sample, capturing the entire tissue thickness. This method is commonly used for smaller lesions or when a representative tissue sample is required (Cancer Research UK, n.d.).

### 1.4.4 Shave Biopsy

A shave biopsy involves the removal of the top layer of the skin or lesion using a specialized instrument. This method is often used for superficial lesions that do not require full-thickness sampling (Cancer Research UK, n.d.).

### 1.4.5 My approach for diagnosis – Deep learning method

By focusing on innovative approaches such as artificial intelligence, this research seeks to contribute to a more accessible skin cancer diagnosis. Many aminoacids are crucial for developing skin cancer and can be detected by electrodes. The data acquired from these aminoacids is analyzed, and the deep learning model will be trained to identify different patterns from each amino acid. Based on the test results, it will indicate the presence or non-presence of these amino acids.

## 1.5 Amino Acids and each role in the development of skin cancer

Traditionally, research on cancer metabolism has primarily focused on central carbon metabolism pathways, such as glycolysis and the tricarboxylic acid (TCA) cycle. However, recent studies have highlighted the critical role of amino acids in cancer metabolism. Beyond their involvement in protein synthesis, amino acids maintain redox balance, regulate energy production, support biosynthetic processes, and preserve cellular homeostasis. These diverse metabolic functions have made amino acid metabolism an increasingly prominent area of interest in cancer research.

### 1.5.1 Glutamine

Glutamine, a non-essential amino acid, supports cell proliferation and survival. It is involved in synthesizing various nutrients and regulates numerous cellular activities. Glutamine deprivation can trigger cell apoptosis, underscoring its significance in cellular metabolism (Filipp et al., 2013).

One of the primary metabolic fates of glutamine is the oxidation of its carbon backbone within the mitochondria. This process provides a significant source of energy for proliferating cells. Through a pathway known as glutaminolysis, glutamine is funneled into the tricarboxylic acid cycle (TCA) to sustain oxidative phosphorylation. Additionally, glutaminolysis may contribute to carbon for fatty acid synthesis. This occurs when the TCA cycle intermediate malate is converted into pyruvate via malic enzyme, followed by acetyl-CoA production from pyruvate-by-pyruvate dehydrogenase.

The metabolic adaptability of cancer cells is exemplified by melanoma, which demonstrates the capacity to utilize glutamine effectively as an alternative carbon source. This metabolic flexibility highlights the critical role of glutamine in the progression of certain cancers (Filipp et al., 2013).

### 1.5.2 Asparagine

Asparagine, an essential amino acid for mammals, is synthesized in multiple organs and plays a crucial role in producing key biomolecules, including glucose, proteins, lipids, and nucleotides. Beyond its metabolic functions, asparagine has been identified as a vital factor in

the growth and development of cancer cells. While many cancer cells can synthesize asparagine, their production often falls short of meeting their metabolic demands (Yuan et al., 2024).

Elevated concentrations of asparagine have been observed in patients with melanoma, highlighting its significance in cancer metabolism. A deficiency in asparagine increases the cellular reliance on glutamine, contributing to glutamine addiction in melanoma cells. Furthermore, asparagine becomes particularly critical under nutrient-deprived conditions, especially without sufficient glutamine, as it helps maintain essential physiological processes (Yuan et al., 2024).

This dual role of asparagine, in both normal metabolic pathways and cancer cell adaptation, underscores its importance as a key target in understanding melanoma and other cancers.

### 1.5.3 Aspartic Acid

Aspartic acid, also referred to as aspartate, plays a pivotal role in the metabolism of cancer cells. Asparagine, an amino acid essential for the survival of cancer cells, relies on precise regulation, a process in which aspartic acid is critically involved. Specifically, aspartic acid serves as a substrate for asparagine synthetase (ASNS), the enzyme responsible for catalyzing asparagine synthesis from aspartic acid and glutamine (Lomelino et al., 2017). This enzymatic reaction highlights the importance of aspartic acid in maintaining the metabolic needs of cancer cells. Therefore aspartic acid abundancy exists in the cell.

These three amino acids are closely interconnected in supporting the growth and proliferation of cancer cells. They function as critical energy sources, ensuring metabolic flexibility by providing fuel for cellular processes, both in the presence and absence of glucose (Lomelino et al., 2017).

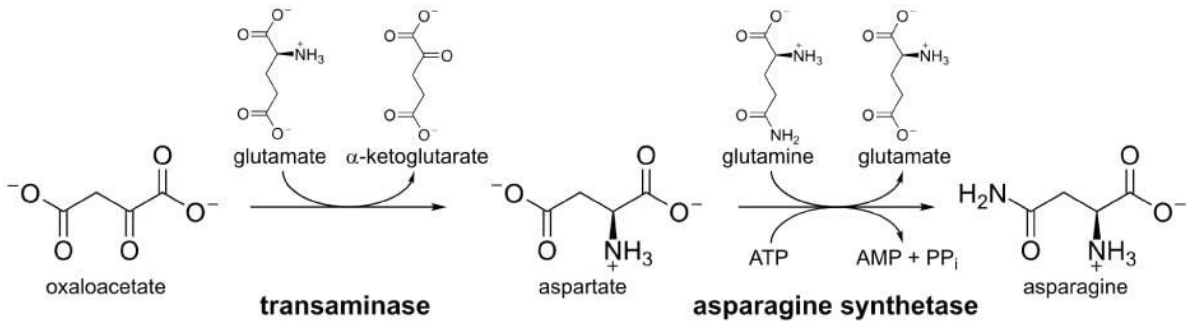


Figure 1.1 Asparagine synthetase

## 1.6 Deep learning models and the importance of them for the future

Deep learning, a subfield of machine learning within artificial intelligence (AI), mimics the human brain's functioning, enabling systems to tackle complex patterns and derive sophisticated insights. Deep learning models autonomously analyze large datasets using neural networks, identifying intricate relationships and adapting without human intervention (NYIT, 2023). This capability drives advancements across applications, including virtual assistants, fraud detection, and autonomous driving.

A deep learning model is structured as interconnected nodes within multiple layers, resembling neural pathways in the human brain. Information flows sequentially through these layers, allowing the network to identify patterns and predict. Deep learning models typically feature at least three layers, enhancing their ability to process unstructured data. Their autonomous learning and adaptability capacity contributes to scalability across domains (AltExSoft, 2023).

Deep learning architectures are specialized for specific challenges. Convolutional Neural Networks (CNNs) excel in image processing tasks, such as diagnostic imaging, while Recurrent Neural Networks (RNNs) handle sequential data like speech recognition and natural language processing (Turing.com, 2023). Long Short-Term Memory (LSTM) networks within RNNs manage tasks involving long-range dependencies, such as language modeling.

Applications span manufacturing, healthcare, aerospace, and consumer electronics. Key applications include:

- **Computer Vision**: Systems interpret and classify images, supporting content moderation and medical diagnostics.
- **Speech Recognition**: Machines process human speech, enabling voice-activated technologies.
- **Recommendation Systems**: Analyze user behavior to provide personalized recommendations on streaming platforms and social media.
- **Natural Language Processing (NLP)**: Enhance machine understanding of human language, with applications in translation and chatbots (LITSLink, 2023).

Deep learning models represent a pivotal development in AI, as they offer sophisticated analytical capabilities that continuously improve through self-directed learning. By leveraging structured architectures tailored to specific tasks, these models drive innovation across various industries.

## 1.7 Metrics for Deep Learning Results

True positives, false positives, true negatives, and false negatives are fundamental components for calculating various metrics to assess model performance. The relevance of specific evaluation metrics depends on several factors, including the nature of the model, the particular task being addressed, the cost associated with different types of misclassifications, and whether the dataset is balanced or imbalanced.

### 1.7.1 Accuracy

Accuracy is a fundamental metric used to evaluate the performance of classification models. It represents the proportion of all positive and negative predictions that are correctly classified. Mathematically, accuracy is defined as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Where:

- TP (True Positives): Correctly predicted positive instances.

- TN (True Negatives): Correctly predicted negative instances.
- FP (False Positives): Incorrectly predicted positive instances.
- FN (False Negatives): Incorrectly predicted negative instances.

An ideal model, with zero false positives and zero false negatives, would achieve an accuracy of 1.0 (100%). As accuracy accounts for all four outcomes from the confusion matrix, it provides a straightforward and intuitive measure of overall model performance.

However, its reliability as a sole evaluation metric diminishes in scenarios involving imbalanced datasets or tasks where the costs of false positives and false negatives differ significantly. For instance, accuracy can be misleading in real-world applications where one class is rare (e.g., detecting fraudulent transactions or diseases). A model predicting the majority class exclusively might achieve high accuracy but fail to identify any instances of the minority class, rendering it ineffective (Google Developers, n.d.).

Therefore, while accuracy is often the default metric for balanced datasets and generic tasks, it should be complemented by more specialized metrics—such as precision, recall, or F1-score—in contexts where dataset imbalance or varying misclassification costs are critical.

### 1.7.2 F1 score

The F1 score is a widely utilized metric for evaluating classification models, particularly in binary and multi-class classification, as well as in large language model (LLM) assessments. It represents the harmonic mean of precision and recall, combining these two metrics into a single value to comprehensively evaluate model performance (V7 Labs, 2023). The F1 score is mathematically defined as:

$$F1\ Score = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

The F1 score can be adapted to prioritize either precision or recall by modifying its weighting parameter, resulting in variations such as F0.5F_{0.5}F0.5, F1F_{1}F1, and F2F_{2}F2. This

flexibility allows for alignment with specific use cases where certain errors (e.g., false positives or false negatives) are more critical than others (V7 Labs, 2023).

A key advantage of the F1 score lies in its effectiveness for imbalanced datasets, where one class significantly outnumbers the others. Unlike accuracy, which only considers the total number of correct predictions, the F1 score accounts for the type of errors made by the model—specifically false positives and false negatives. This characteristic makes it particularly relevant in applications such as fraud detection, medical diagnostics, and other domains where errors have significant consequences (V7 Labs, 2023).

In summary, the F1 score is essential for evaluating classification models, especially in scenarios involving imbalanced datasets or when the costs of false positives and negatives differ (V7 Labs, 2023).

### 1.7.3 Precision

Precision is a key metric used to evaluate the performance of a machine learning model, explicitly assessing the quality of optimistic predictions. It is defined as the proportion of true positive predictions out of all positive predictions made by the model, including both true positives and false positives. Mathematically, precision is expressed as:

$$Precision = \frac{TP}{TP + FP}$$

Precision quantifies the reliability of a model's positive predictions. A precision value of 1.0 (or 100%) indicates that all positive predictions made by the model were correct, reflecting an absence of false positives. However, achieving perfect precision in real-world applications is highly unlikely due to inherent trade-offs with other metrics, particularly recall (Google Developers, n.d.).

Precision and recall often exhibit an inverse relationship. As precision improves (fewer false positives), recall typically decreases, as some true positives may be missed in favor of higher specificity. Conversely, improving recall (fewer false negatives) may reduce precision due to an increase in false positives. This trade-off is particularly evident when adjusting the

classification threshold of a model. For example, increasing the threshold can reduce false positives and improve precision but may increase false negatives, lowering recall (Google Developers, n.d.).

Precision is most meaningful in scenarios where false positives carry significant consequences, such as in medical diagnostics or fraud detection. However, its utility diminishes when dealing with highly imbalanced datasets, where the number of positive instances is extremely low. In such cases, a model may achieve high precision without being practically useful, as it may fail to identify the majority of true positives (Google Developers, n.d.).

In summary, precision is a critical metric for evaluating the accuracy of a model's positive classifications, particularly in applications where minimizing false positives is a priority. However, it must be interpreted alongside other metrics, such as recall and the overall context of the problem, to ensure a balanced assessment of model performance.

### 1.7.4 Recall

Recall, also known as the true positive rate (TPR), is a crucial metric in classification tasks. It represents the proportion of actual positive instances that are correctly identified as positive by the model. Mathematically, recall is defined as:

$$Recall = \frac{TP}{TP + FN}$$

where TPTPTP stands for true positives, and FNFNFN refers to false negatives. False negatives are actual positives misclassified as negative, which explains their inclusion in the denominator (Google Developers, n.d.).

In imbalanced datasets, where the number of positive instances is minimal, recall may not fully indicate model performance, as it could be skewed by the small number of positives (Google Developers, n.d.).

### 1.7.5 Cohen's kappa

Cohen's kappa (κ) is a statistical measure used to evaluate the agreement between two classifiers, accounting for the possibility of agreement occurring by chance. It is beneficial in classification tasks where the data may exhibit class imbalances. The formula for Cohen's kappa is given by:

$$\kappa = \frac{Po - Pe}{1 - Pe}$$

where P0 represents the observed accuracy of the model, and Pe denotes the expected accuracy, which is the probability of the model's predictions agreeing with the actual class values by chance (Towards Data Science, 2021).

Cohen's kappa corrects for the agreement that might occur by chance, offering a more reliable measure of the model's performance. Unlike simple accuracy, which imbalanced class distributions can bias, Cohen's kappa adjusts for this by considering how often the classifier's agreement with actual outcomes is better than random chance. Therefore, it provides a more robust evaluation metric, especially when dealing with datasets with significant class imbalances. In summary, Cohen's kappa is an essential tool for assessing classification models, particularly in scenarios where traditional accuracy may not fully reflect model performance due to random chance or class imbalances. By considering both the observed agreement and the expected agreement, it offers a more nuanced understanding of the classifier's true performance (Towards Data Science, 2021).

### 1.7.6 AUC – ROC

The Area Under the Receiver Operating Characteristic (AUC - ROC) curve is a widely used performance metric in classification problems, particularly for evaluating models at various threshold settings. The ROC curve is a probability curve, and the AUC represents the degree of separability between the two classes. It quantifies how well the model distinguishes between the positive and negative classes. A higher AUC indicates that the model is better at correctly

classifying the positive class (1) as positive and the negative class (0) as negative. For instance, in a medical context, a higher AUC implies that the model can more effectively distinguish between patients with the disease and those without.

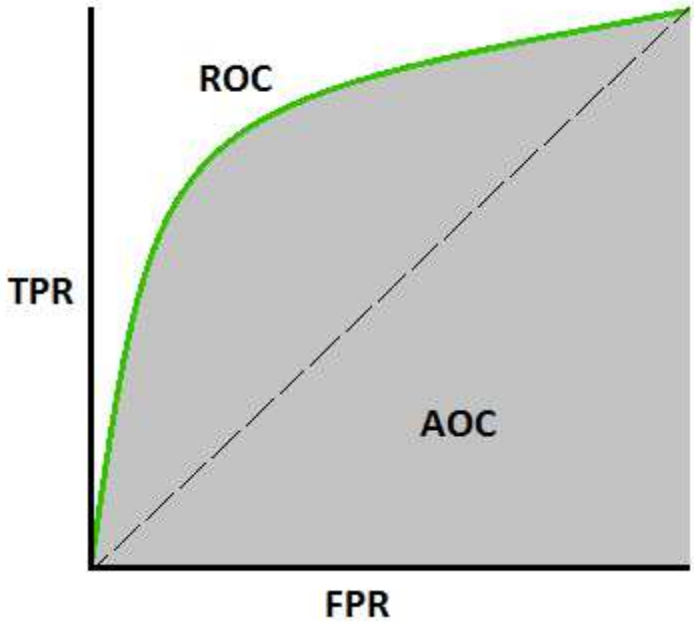


Figure 1.2 AUC-ROC curve

In contrast, when the AUC is close to 0, the model is essentially making predictions that are reversed, or reciprocated—classifying the negative class as positive and vice versa. An AUC of 0 indicates complete misclassification, where the model consistently predicts the opposite class.

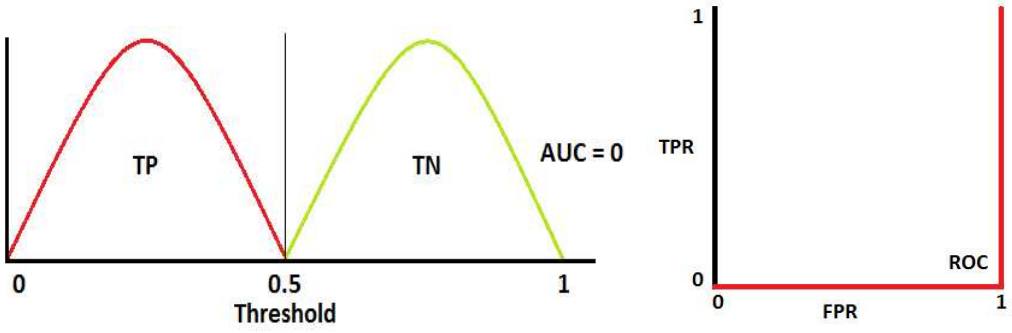


Figure 1.3 Graph of AUC=0

AUC values close to 0.5 suggest that the model has no discriminative ability, performing no better than random chance.

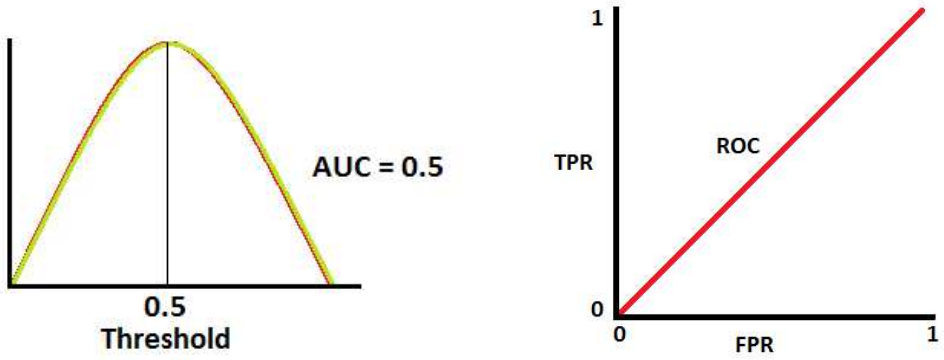


Figure 1.4 Graph of AUC=0.5

The ROC curve is plotted with the True Positive Rate (TPR) on the y-axis and the False Positive Rate (FPR) on the x-axis. The TPR, also known as recall, reflects the proportion of true positives that are correctly identified by the model, while the FPR measures the proportion of false positives. A model with perfect classification would have an AUC close to 1.0, corresponding to a curve that reaches the top-left corner of the plot. This indicates ideal separability, where the model can perfectly distinguish between classes (Towards Data Science, 2020).

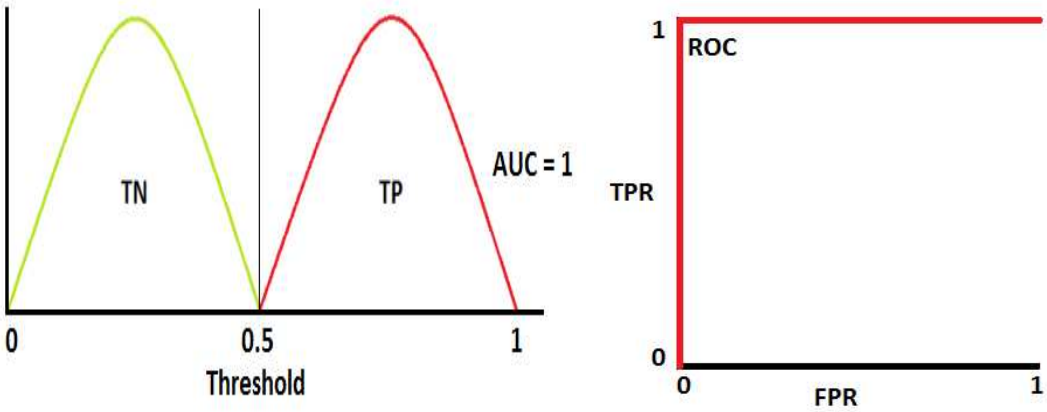


Figure 1.5 Graph of AUC=1

To interpret these values further, when the ROC curves of the two classes do not overlap, the model demonstrates an ideal separability measure, as it can ideally differentiate between the positive and negative classes. However, in real-world scenarios, some overlap in the distributions of class probabilities is inevitable, leading to Type I and Type II errors. The threshold for classification can be adjusted to minimize or maximize these errors (Towards Data Science, 2020).

An AUC of 0.7, for example, indicates a 70% chance that the model will correctly distinguish between a randomly chosen positive instance and a randomly chosen negative instance. This level of AUC represents a model that is performing reasonably well but still has room for improvement.

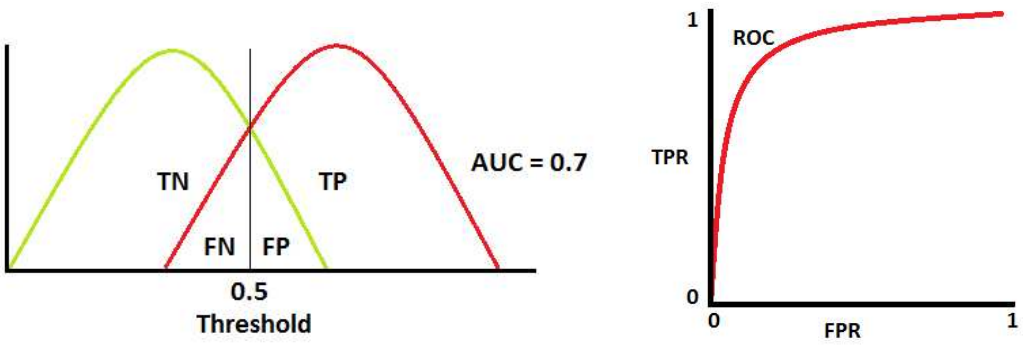


Figure 1.6 Graph of AUC=0.7

## 1.8 Aim

The principal aim of this project is to investigate a different approach to early skin cancer diagnosis using the data of amino acids related to skin cancer, measured through different electrodes and tested in a artificial intelligence model of deep learning.

## 1.9 Research questions

1) What are the specific amino acid biomarkers associated with different types of skin cancer, and how can they be accurately measured using electrode-based sensors?

2) Which deep learning algorithms are most effective?

3) What are the potential challenges and limitations in implementing electrode-based amino acid measurement and deep learning models in a clinical setting for early skin cancer diagnosis?

# 2 Methods

## 2.1 Amino acid mixtures

This study carefully selected three amino acids based on their relevance to skin cancer: asparagine, aspartic acid, and glutamine. These amino acids play a significant role in promoting the growth and proliferation of cancer cells.

Each amino acid was dissolved in phosphate-buffered saline (PBS) at the same concentration. Measures were taken to maintain a controlled environment and to protect the samples from external factors that could compromise their stability. The amino acids were prepared individually and stored in separate containers to avoid any risk of cross-contamination.

## 2.2 Procedures and use of the amino acid

The initial step of this study involved using small quantities of each individual amino acid to collect data specific to their individual effects. In the second step, the amino acids were combined in pairs to evaluate whether the model's behaviour aligned with theoretical predictions. Finally, in the third step, equal amounts of all three amino acids were mixed in a single container to obtain the final results for the experiment.

## 2.3 E-tongue

Electronic tongue (ET) is an analytical tool that can be successfully used when the purpose of the analysis is to assign a sample to a specific class, identify the compounds of the sample, as well as for quantitative analysis. Over the last years this field was extensively explored, hence numerous applications of ET systems for food quality control, environmental monitoring, medical diagnosis, pharmaceutical analysis or bioprocess control was reported (Falk et al., 2021).

Moreover, sensors selected to co-create ET sensor array may differ in their design and signal generation method, therefore arrays consisting of electrochemical (voltammetric and

potentiometric), optical, mass sensors, as well as biosensors can be distinguished. However, most of ET systems are based on electrochemical sensors (Falk et al., 2021).

## 2.4 Data acquisition and data treatment for testing

### 2.4.1 Electrochemical sensors

Electrochemical sensors are devices that quantitatively detect specific chemical species through oxidation or reduction currents. These sensors offer notable advantages, including simple measurement procedures, fast response times, and sufficient sensitivity and selectivity. While traditional chemical sensors that use indirect detection methods (e.g., UV-vis spectroscopy) can achieve high sensitivity, they are less suitable for *in situ* measurements. In contrast, electrochemical sensors can continuously monitor concentration changes over time (Hu, Yoon & Park, 2020).

To enhance the performance of electrochemical sensors, modifications to the electrode surface are essential. Electrocatalytic materials are commonly applied to improve sensitivity, while coatings with specific films, enzymes, or mediators are employed to achieve high selectivity (Hu, Yoon & Park, 2020).

For this study it was used a basic electronic setup, the surfaces of electrodes were not modified. The purpose of this is to test the possibility of implementing this electronic sensors in the deep learning method to identify the amino acids and to evaluate the data acquired from each electrode. It was used three different electrodes composed each one by gold, platinum and carbon.

### 2.4.2 Square wave voltammetry

Square wave voltammetry (SWV) is a valuable electrochemical technique commonly employed by researchers to study redox reactions. Among pulsed techniques, SWV most closely resembles the output of cyclic voltammetry, and it offers the advantage of minimizing the capacitive background signal (Macias Sensors, n.d.).

SWV belongs to the category of pulse voltammetry methods, alongside normal pulse voltammetry and differential pulse voltammetry. It stands out for its ability to reduce the capacitive background current at the working electrode, while enhancing the signal-to-noise

ratio for Faradaic currents associated with oxidation and reduction events (Macias Sensors, n.d.).

The excitation waveform in SWV is a modified linear sweep voltammetry, incorporating a square wave superimposed on a staircase potential. This approach applies two pulses to the working electrode at each step of the staircase: one in the forward direction and one in the reverse. This unique waveform design enables two current sampling points, which can be processed to minimize capacitive current contributions (Macias Sensors, n.d.).

Additionally, SWV minimizes background capacitive currents, yielding an output similar to that of differential pulse voltammetry. By subtracting the reverse current from the forward current, SWV produces a single peak per oxidation/reduction pair on a stable baseline. The quality of this baseline, the signal-to-noise ratio, and the width of the peaks are highly influenced by the parameters set during data acquisition, necessitating careful optimization for each application.

### 2.4.3 Spectograms

Spectrograms are widely used in fields such as seismology, linguistics, sonar, radar, and speech processing to analyze both audio and other signals. Different instruments produce spectrograms with distinct frequency patterns, allowing differentiation. Similarly, spectrograms generated from amino acids exhibit unique frequency characteristics, which may serve as a basis for their identification. However, these differences are often imperceptible to the human eye.

To computationally analyze these spectrograms, we developed a Python-based approach using the SciPy signal processing library. The function specto() applies the Short-Time Fourier Transform (STFT) via signal.spectrogram(), converting an input signal into its time-frequency representation. The power spectral density (PSD) of the signal is then transformed into a logarithmic scale (dB) to enhance visibility.

The generated spectrograms reveal distinct patterns for each amino acid, but manual identification remains challenging. By extracting these spectrograms and feeding them into a deep learning model, it may be possible to train an algorithm to automatically recognize amino acids in mixed solutions. Below is the Python function used for spectrogram computation and visualization:

```
import numpy as np
from scipy import signal
import matplotlib.pyplot as plt


# Function to compute and display the spectrogram
def specto(dp):
    fs = 20
    f, t, Sxx = signal.spectrogram(dp, fs, nperseg=54)
    Sxx_db = 10 * np.log10(Sxx)


    plt.pcolormesh(t, f, Sxx_db, shading='gouraud', cmap='viridis')
    plt.ylabel('Frequency [Hz]')
    plt.xlabel('Time [sec]')
    plt.show()


dospect = specto(d)
```

Listing 2.1 Python function for spectrogram computation and visualization.

## 2.5 Deep learning models and optimizers

In the first section of the code below, all necessary libraries and modules required for loading the model and generating predictions were imported. These imports were organized according to their functionality:

- **General Purpose Libraries:**
    - **os:** Used for interacting with the operating system (e.g., handling file paths).
    - **numpy:** Used for numerical computations and efficient array manipulations.
- **Visualization:**
    - **matplotlib.pyplot:** Used for plotting graphs and visualizing data outputs.

- **Image Preprocessing and Deep Learning Tools:**
    - **keras.preprocessing.image:** Provides tools to load and process image data.
    - **keras.applications.nasnet:** Imports the pre-trained NASNetLarge model and its associated preprocess_input function, which are used for feature extraction from images.
    - **keras.models.Sequential** and **keras.layers** (including **Dense** and **Flatten**): Used to construct a simple neural network model.
    - **keras.optimizers.Adam** and **RMSprop:** Optimizers used to facilitate efficient gradient descent during model training.
- **Data Preparation:**
    - **sklearn.model_selection.train_test_split:** Used to split the dataset into training and testing subsets.
    - **tensorflow.keras.utils.to_categorical:** Converts class vectors (integers) into binary class matrices, which is essential for multi-class classification tasks.

This section establishes the foundation by ensuring that all required tools are available for subsequent tasks such as feature extraction, model construction and training, and data visualization.

```
import os
import numpy as np
import matplotlib.pyplot as plt
from keras.preprocessing import image
from keras.applications.nasnet import NASNetLarge, preprocess_input
from keras.models import Sequential
from keras.layers import Dense, Flatten
from keras.optimizers import Adam, RMSprop
from sklearn.model_selection import train_test_split
from tensorflow.keras.utils import to_categorical
```

Listing 2.2 Imported libraries

### 2.5.1 NasNetLarge

NASNet-Large is a convolutional neural network trained on more than a million images from the ImageNet database. This network has been designed to learn rich feature representations across various image types, making it highly effective for transfer learning applications (Keras, n.d.).

In the following part of the code, the pre-trained NASNetLarge model (with ImageNet weights and without its top classification layers) is loaded to extract high-level features from the input images. The training and testing datasets are first preprocessed using the model's dedicated function to ensure the images are correctly formatted. The preprocessed images are then passed through the model to generate feature representations for both datasets. Additionally, predictions are generated on the training data for further analysis.

```
# Load the pre-trained NASNetLarge model

model1 = NASNetLarge(weights='imagenet', include_top=False,
input_shape=(224, 224, 3))


# Preprocess the training and testing data.

x_train_norm = preprocess_input(np.array(x_train))

x_test_norm = preprocess_input(np.array(x_test))


# Use NASNetLarge to extract features from the preprocessed images.

train_features = model1.predict(x_train_norm)

test_features = model1.predict(x_test_norm)


# Generate predictions on the training data (for demonstration or
further analysis).

preds = model1.predict(x_train_norm)
```

Listing 2.3 Loading the pre-trained NASNetLarge model

### 2.5.2 Activation functions

Activation functions are critical components in neural networks and deep learning models, activating hidden nodes in a manner that optimizes output quality. The primary function of an activation layer is to introduce nonlinearity into the model, enabling it to capture complex patterns and relationships within the data, which would be unattainable in a purely linear model.

In this section, a simple fully-connected neural network is constructed to accept the extracted features as input for classification. First, the input features are flattened to remove the spatial dimensions. Next, a dense layer with 1024 neurons and ReLU activation is added to learn complex representations. Finally, an output layer with three neurons and softmax activation is incorporated to produce a probability distribution across the three classes. The model is then compiled using the Adam optimizer and categorical cross-entropy loss, and the model summary is displayed to review its architecture..

```
# Build a simple fully-connected model for classification.

# The model takes extracted features as input.

model = Sequential()

# Flatten the input features (excluding the first dimension which is the
batch size)

model.add(Flatten(input_shape=train_features.shape[1:]))

# Add a fully connected layer with 1024 neurons and ReLU activation.

model.add(Dense(1024, activation='relu'))

# Output layer: 3 neurons for 3 classes, using softmax for probability
distribution.

model.add(Dense(3, activation='softmax'))

# Compile the model with Adam optimizer, categorical cross-entropy loss,
and accuracy as metric.

model.compile(optimizer='adam', loss='categorical_crossentropy',
metrics=['accuracy'])

# Display the model summary.

model.summary()
```

Listing 2.4 Fully connecting the model for classification

### 2.5.3 Sofmax

Many multi-layer neural network architectures conclude with a penultimate layer that outputs raw, real-valued scores. These unscaled scores, however, may present practical challenges in terms of interpretability and usability for downstream processes. Applying a softmax function at the final layer effectively addresses this issue by transforming these scores into a normalized probability distribution. This conversion enables more intuitive interpretation by end users and facilitates compatibility with subsequent system inputs. As such, it is common practice to include a softmax layer as the terminal layer of the network (Phillips, 2020).

### 2.5.4 Relu

The Rectified Linear Unit (ReLU) is a widely utilized non-linear activation function in deep neural networks, also referred to as the rectifier activation function. ReLU introduces nonlinearity into the neural network, essential for addressing the vanishing gradient problem frequently encountered during model training. This function thus enables deep networks to capture and learn complex, non-linear relationships within data (Builtin, n.d.).

Mathematically, the ReLU function outputs the input value when the input is positive and zero when the input is negative. This simplicity in formulation contributes to its computational efficiency and widespread adoption in deep learning applications (Builtin, n.d.).

### 2.5.5 Adam

Adaptive Moment Estimation (Adam) is an optimization algorithm commonly employed to enhance the efficiency of gradient descent in neural network training. This method proves particularly advantageous in large-scale problems involving extensive datasets or numerous parameters, as it is both memory-efficient and computationally effective. Adam integrates aspects of the 'gradient descent with momentum' approach and the Root Mean Square Propagation (RMSProp) algorithm, combining momentum-based gradient updates with adaptive learning rates. This hybridization allows Adam to accelerate convergence and improve stability in training deep learning models (GeeksforGeeks, n.d.).

The Adam optimizer integrates two key methodologies from gradient descent optimization: Momentum and RMSProp. Momentum accelerates the gradient descent process by incorporating an ‘exponentially weighted average’ of the gradients from previous steps. By smoothing the gradients this way, momentum helps the optimizer converge more quickly and consistently towards the minimum.

RMSProp is an adaptive learning rate method designed to address limitations in Adaptive Gradient Algorithm (AdaGrad). Unlike AdaGrad, which accumulates the sum of squared gradients over time, RMSProp employs an ‘exponential moving average’ of squared gradients. This approach adapts the learning rate dynamically, allowing the algorithm to maintain efficient updates without slowing excessively, even in prolonged training scenarios (GeeksforGeeks, n.d.).

### 2.5.6 Categorical cross entropy

In classification tasks, each instance belongs to a known class label with a probability of 1.0, while all other class labels are assigned a probability of 0. The model's objective is to estimate the probability that a given instance belongs to each possible class. To measure the accuracy of these predictions, cross-entropy loss is commonly employed to assess the divergence between predicted and true class probabilities (V7 Labs, 2024).

During training, the predicted probability for each class is compared to the expected binary output (0 or 1). The cross-entropy loss function penalizes discrepancies between these predictions and the target values, with a logarithmic penalty applied according to the magnitude of the difference: larger penalties are applied for significant deviations near 1, and smaller penalties for minor discrepancies close to 0. This scoring mechanism enables the model to adjust its weights iteratively, with the objective of minimizing the cross-entropy loss. A lower loss value indicates a model with better predictive performance (V7 Labs, 2024).

### 2.5.7 Batch size

Models trained with smaller batch sizes generally do well on training and for validation. On the other hand, it takes more time to train and validate the data compared to higher batch sizes. Results show that the batch size of 32 gives the best results out of all the batch sizes. On the contrary a batch size of 2048 gives the worst results (Thakur, 2025).


Figure 2.1 Test error rate and time taken for different batch sizes during neural network training. (Adapted from Ayush Thakur, 2025).

In this step, the fully-connected model is trained using the extracted training features and the corresponding one-hot encoded labels. A portion of the data is set aside as a validation set to monitor the model's performance during training. The model is trained with a batch size of 32 for 18 epochs, and the training history is stored for subsequent analysis.

```
hist = model.fit(train_features, y_train_encoded,
validation_data=(test_features, y_test_encoded), batch_size=32,
epochs=18)
```

Listing 2.5 Training the model with a batch size of 32 and 18 epochs

# 3 Results and Discussion

This study contains a total 1350 samples of data, 150 samples for each amino acid and the corresponding electrode. The samples were converted into spectrograms in order to be read by the deep learning model and then trained.

Each training was done separately for each electrode, the three amino acids samples from a specific electrode served as the input for the deep learning model.

A graph of training and validation accuracy and its corresponding training and validation loss were put together. Both represent the model's behavior throughout the epochs that it was learning. A heat map was built to show the model prediction for each amino acid and the associated electrode.

Therefore, it is shown below the corresponding graphs and heat map for each electrode. The results are divided into 3 separate sections to study the difference between each electrode and their behaviour with the deep learning model.

The heatmaps below illustrate the predictions the model and the corresponding electrode made. With 10% of the available data allocated for validation and 90% for training, the prediction values range from 0 to 15. A value of 0 indicates that no samples were correctly predicted for the corresponding amino acid, while a value of 15 signifies perfect prediction accuracy for all validation samples of that amino acid.

## 3.1 Gold Electrode

The model was trained using 150 samples of each amino acid measured with a gold electrode.

Results showed an accuracy of 82%, this value is considered valuable and realistic since its between 70% and 90%. This can be concluded by paying attention to the training and validation accuracy in the graph below. Throughout the epochs it can be seen that the training and validation increase significantly, meaning that the model was showing good results of training. It was also used other types of metrics, the F1 score is showed 81%, precision of 84%, recall of 82,2%, Cohens kappa of 73,3% and a ROC AUC of 93%. All this values are between the range of 70% and 90% indicating the reliability of the model.

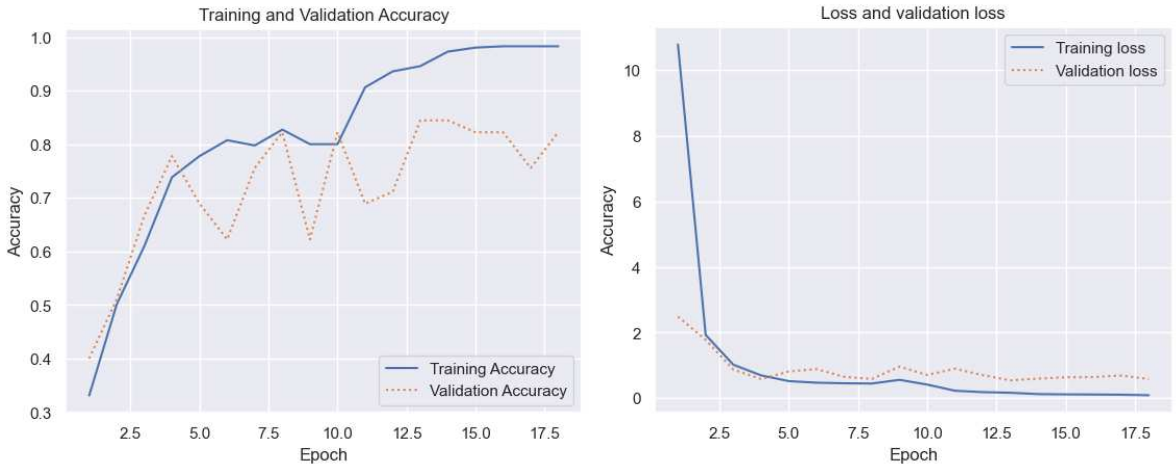


Figure 3.1 Graph of Training and validation accuracy. Graph of loss and validation loss – Gold electrode

For the gold electrode, the model performed particularly well in identifying aspartic acid and glutamine, achieving 15 and 14 correct predictions, respectively. These results suggest that these two amino acids exhibit distinct patterns that the model could effectively differentiate, with the gold electrode capturing these differences accurately. Conversely, the predictions for asparagine were less accurate, with only 8 out of 15 samples correctly classified. The model appears to confuse asparagine with the other amino acids, highlighting a potential area for improvement in distinguishing its spectral features.

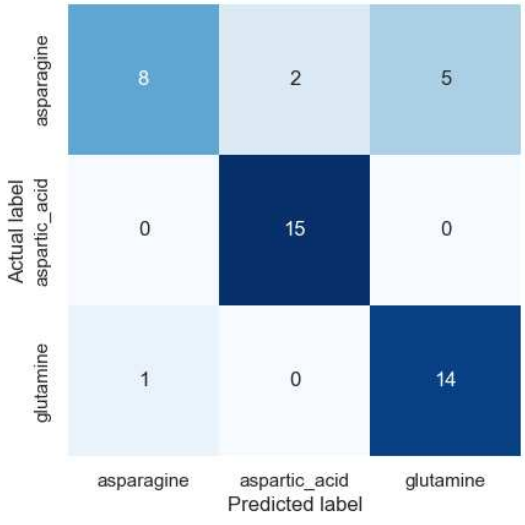


Figure 3.2 Heat map – Gold electrode

## 3.2 Platinum Electrode

For this electrode, results showed that the model had 80% of accuracy. This value of accuracy means that the model is valuable and realistic since it is also in the range of 70% to 90%. As it is shown below in the graph of training and validation accuracy, the model had a good learning curve throughout the epochs. The results for the other parameters also showed good signs for the model training and validation since it got an F1 score of 80%, precision of 80%, a recall of 80%, a Cohens Kappa of 70% and a ROC AUC of 88%. Similar to the gold electrode, the platinum electrode also show good results for the model prediction.

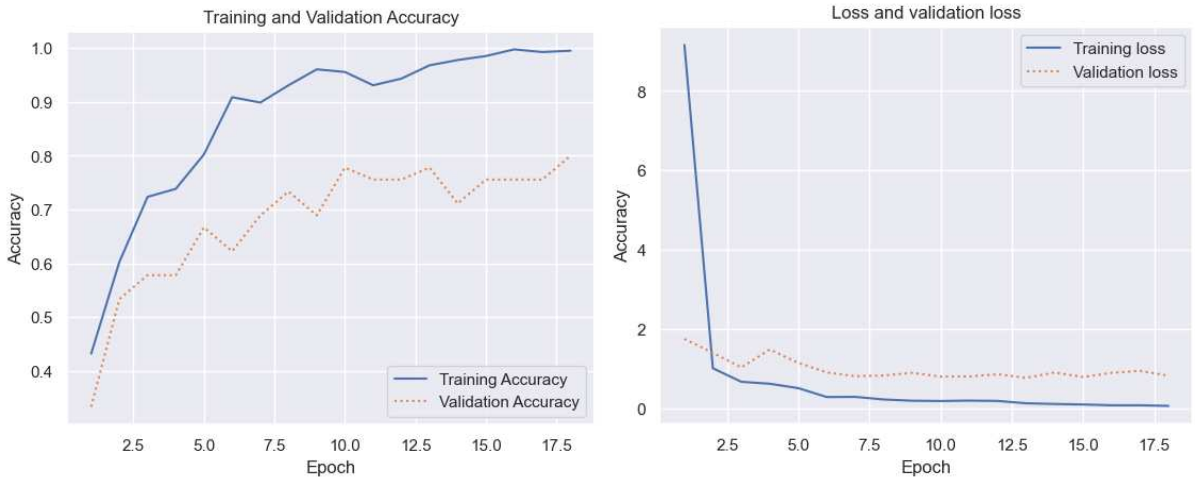


Figure 3.3 Graph of Training and validation accuracy. Graph of loss and validation loss – Platinum electrode

While the heatmap for the gold electrode demonstrated strong performance, the results for the platinum electrode are also noteworthy, particularly in the prediction of aspartic acid, which was classified with perfect accuracy (15 out of 15 samples correctly predicted). However, for the other amino acids, the model displayed minor challenges in achieving full accuracy. Specifically, 10 out of 15 samples labeled as asparagine were correctly identified, and 11 out of 15 glutamine samples were accurately classified. These results indicate that while the model performs well overall, there is room for improvement in distinguishing asparagine and glutamine more reliably.

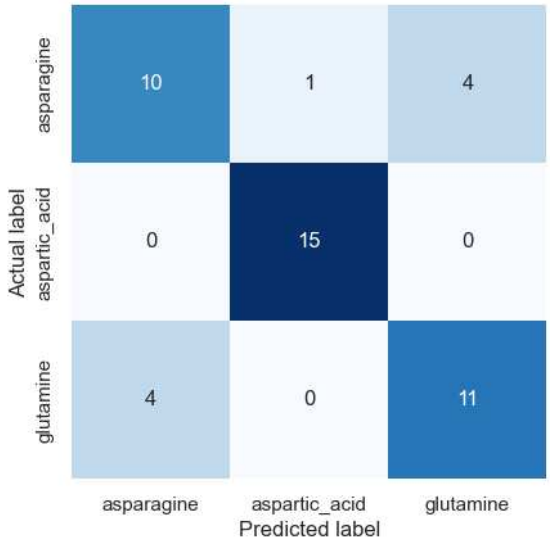


Figure 3.4 Heat map – Platinum electrode

## 3.3 Carbon electrode

The carbon electrode revealed slightly worse results than the previous electrodes. Below it can be seen an accuracy of 78% and this results can be strengthened by looking at the loss and validation loss that had much higher values than the previous graphs. The lower the validation loss the better the model is able to learn and make the correct predictions during each epochs. The other metrics of this electrode were 78.2% for the F1 score, precision of 79%, recall of 78%, Cohens kappa of 67% and ROC AUC of 91%.


Figure 3.5 Graph of Training and validation accuracy. Graph of loss and validation loss – Carbon electrode

Interestingly, the heatmap for the carbon electrode highlights a distinct behavior compared to the other electrodes. The model performed particularly well in predicting asparagine, with 13 out of 15 samples correctly classified. However, for aspartic acid and glutamine, the model exhibited slightly lower accuracy, with 12 and 10 correct predictions, respectively.

When comparing this to the other electrodes, a shared characteristic emerges: the values in the heatmap deviate slightly from the total number of samples tested (15 per amino acid). This suggests that, for both electrodes, the model faces minor challenges in extracting robust patterns from the spectrograms, which affects its ability to classify the amino acids perfectly.

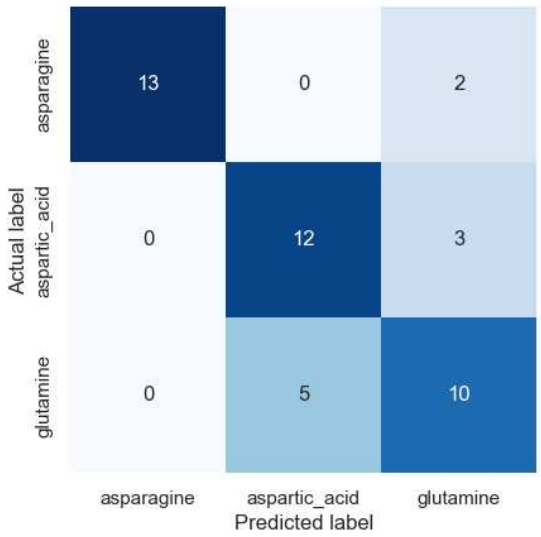


Figure 3.6 Heat map – Carbon electrode

## 3.4 Model prediction

To evaluate the model's ability to make accurate predictions, a final test was conducted using a mixture of the three amino acids at equal concentrations. Each mixture was applied to the surface of the respective electrodes, following the same procedure as the previous measurements with individual amino acids.

### 3.4.1 Mixture of the 3 amino acids on the gold electrode

The results for this final test were analyzed individually for each electrode. Based on the earlier findings, it was anticipated that the gold electrode would yield the most accurate predictions when the model was tested with data generated from the amino acid mixture. The model

predicted the mixture as follows: 16% as asparagine, 69% as aspartic acid, and 15% as glutamine. These results indicate that the model successfully detected the presence of all three amino acids in the solution, with aspartic acid being the amino acid most easily identified by the model.

### 3.4.2 Mixture of the 3 amino acids on the platinum electrode

The predictions for the platinum electrode were less accurate than anticipated, considering the promising results observed during the training phase. Specifically, the model predicted 93% as asparagine, 0% as aspartic acid, and 7% as glutamine. These results deviate significantly from the expected proportions for this experiment. Unlike the gold electrode, the model failed to detect the presence of one amino acid, as aspartic acid was assigned a prediction of 0%. Furthermore, there was a substantial discrepancy between the predictions for glutamine (7%) and asparagine (93%), with the latter being the most accurately identified amino acid in the solution.

### 3.4.3 Mixture of the 3 amino acids on the carbon electrode

The results for the carbon electrode presented a unique outcome compared to the other electrodes. Although the carbon electrode exhibited the lowest accuracy during the training phase (78%)—compared to 80% and 82% for the platinum and gold electrodes, respectively—the predictions in this test outperformed those of the platinum electrode. The model predicted 0% as asparagine, 53% as aspartic acid, and 47% as glutamine. Similar to the platinum electrode, one amino acid (asparagine) was not detected in the solution. However, the predicted percentages for aspartic acid and glutamine were relatively close, suggesting that the model was capable of reliably distinguishing these two amino acids in the mixture.

# 4 Conclusions

To conclude this method shows a good curve of improvement. Enabling more data from each amino acid or adding other amino acids to this study may increase the length of data the model will train and therefore have better results. Giving more data to the model will also give more differentiation and will give better predictions to know if a mole on the body is cancerous or not for example. Based on the number of samples and the results shown in this study, it is visible that with more data, the model's accuracy could reach numbers higher than 90%.

To sum up, I believe meeting these requirements would transform this concept into something useful in the future. The innovation of using AI to identify diseases could have a major impact in the medicine industry. AI is able to identify images and find patterns that the human eye cant see. This could enable the creation of devices that could detect this problems in early stages minimizing the risk of skin cancer development.

# 5 Future work

While this study demonstrated the feasibility of using Square Wave Voltammetry combined with deep learning for amino acid detection in skin cancer diagnosis, several avenues for improvement and further research remain.

**Expansion of Amino Acid Analysis:** Future studies should incorporate a broader range of amino acids and other relevant biomarkers linked to skin cancer progression. This could enhance the specificity and sensitivity of the diagnostic model.

**Optimization of Electrodes:** The results showed that electrode material significantly influenced detection accuracy. Investigating novel materials, such as nanostructured surfaces or hybrid composites, could improve signal quality and selectivity.

**Integration with Real Biological Samples:** To bridge the gap between laboratory experiments and clinical applications, future research should focus on testing the method with real biological samples, such as skin tissue extracts or patient serum, to validate the approach in physiologically relevant conditions.

**Enhancement of the Deep Learning Model:** Improving the model architecture, incorporating more advanced techniques and increasing the dataset size could further boost classification performance and generalization.

By addressing these aspects, future work can refine and extend the applicability of this approach, bringing it closer to practical use in early skin cancer diagnosis.

## References


AltExSoft Deep Learning Explained. Available at: https://www.altexsoft.com/blog/deep-learning/ (Accessed: 2024).

American Academy of Dermatology Common types of skin cancer. [online] Available at: https://www.aad.org/public/diseases/skin-cancer/types/common (Accessed: 2024).

Builtin ReLU activation function: What it is & how it works. Available at: https://builtin.com/machine-learning/relu-activation-function (Accessed: 2024).

Cancer Research UK Tests for skin cancer. [online] Available at: https://www.cancerresearchuk.org/about-cancer/skin-cancer/getting-diagnosed/tests (Accessed: 2024).

Chiu, M., Taurino, G., Bianchi, M.G., Kilberg, M.S. and Bussolati, O. (2020) Asparagine synthetase in cancer: Beyond acute lymphoblastic leukemia. Frontiers in Oncology, 9, p.1480.

Darowicki, K., Zielinski, A., Mielniczek, M., Janicka, E. and Gawel, L. (2021) Polynomial description of dynamic impedance spectrogram—Introduction to a new impedance analysis method. Electrochemistry Communications, 129, p.107078.

Falk, M., Nilsson, E.J., Cirovic, S., Tudosoiu, B. and Shleev, S. (2021) Wearable electronic tongue for non-invasive assessment of human sweat. Sensors, 21(21), p.7311.

Filipp, F.V., Ratnikov, B., De Ingeniis, J., Smith, J.W., Osterman, A.L. and Scott, D.A. (2013a) Glutamine-fueled mitochondrial metabolism is decoupled from glycolysis in melanoma. Pigment Cell Melanoma Research, 25(6), pp.732–739.

Filipp, F.V., Ratnikov, B., De Ingeniis, J., Smith, J.W., Osterman, A.L. and Scott, D.A. (2013b) Glutamine-fueled mitochondrial metabolism is decoupled from glycolysis in melanoma. Pigment Cell Melanoma Research, 26(4), pp.553–563.

GeeksforGeeks Adam optimizer. Available at: https://www.geeksforgeeks.org/adam-optimizer/ (Accessed: 2024).

Google Developers Classification: Accuracy, Precision, and Recall. Available at: https://developers.google.com/machine-learning/crash-course/classification/accuracy-precision-recall (Accessed: 2024).

Gorgoglione, R., Impedovo, V., Riley, C.L., Fratantonio, D., Tiziani, S., Palmieri, L., Dolce, V. and Fiermonte, G. (2022) Glutamine-derived aspartate biosynthesis in cancer cells: Role of mitochondrial transporters and new therapeutic perspectives. Cancers, 14(1), p.245.

Hu, Y., Yoon, J. and Park, J.-H. (2020) Wearable and implantable electrochemical sensors for healthcare monitoring. Biosensors and Bioelectronics, 170, p.112684.

Jia, Y., Sun, C., Song, W., Xu, C. and Yu, J. (2020) Advances in applications of intrinsic fluorescence for studying native structures of proteins. Sensors and Actuators B: Chemical, 330, p.129389.

Keras NASNet - Applications. Available at: https://keras.io/api/applications/nasnet/ (Accessed: 2024).

Lieu, E.L., Nguyen, T., Rhyne, S. and Kim, J. (2020) Amino acids in cancer. Experimental & Molecular Medicine, 52(1), pp.15–30.

LITSLink  Future of Deep Learning Explained. Available at: https://litslink.com/blog/future-of-deep-learning-explained (Accessed: 2024).

Lomelino, C.L., Andring, J.T., McKenna, R. and Kilberg, M.S. (2017) Asparagine synthetase: Function, structure, and role in disease. The Journal of Biological Chemistry, 292(49), pp.19952–19958.

NYIT Deep Learning and Neural Networks: The Future of Machine Learning. Available at: https://online.nyit.edu/blog/deep-learning-and-neural-networks-the-future-of-machine-learning (Accessed: 2024).

Phillips, H.J. A simple introduction to softmax. Available at: https://medium.com/@hunter-j-phillips/a-simple-introduction-to-softmax-287712d69bac (Accessed: 2024).

Stollnitz, E. Spectrograms and scaleograms. Available at: https://bea.stollnitz.com/blog/spectrograms-scaleograms/ (Accessed: 2024).

Thakur, A. What's the Optimal Batch Size to Train a Neural Network? Available at: https://wandb.ai/ayush-thakur/dl-question-bank/reports/What-s-the-Optimal-Batch-Size-to-Train-a-Neural-Network---VmlldzoyMDkyNDU (Accessed: 2024).

Turing.com Understanding the Future of Deep Learning. Available at: https://www.turing.com/kb/understanding-the-future-of-deep-learning (Accessed: 2024).

Towards Data Science Understanding AUC-ROC Curve. Available at: https://towardsdatascience.com/understanding-auc-roc-curve-68b2303cc9c5 (Accessed: 2024).

Towards Data Science Cohen's Kappa: What It Is, When to Use It, and How to Avoid Its Pitfalls. Available at: https://towardsdatascience.com/cohens-kappa-what-it-is-when-to-use-it-and-how-to-avoid-its-pitfalls-e42447962bbc (Accessed: 2024).

V7 Labs F1 Score Guide: How to Measure Classification Performance. Available at: https://www.v7labs.com/blog/f1-score-guide (Accessed: 2024).

V7 Labs Cross-Entropy Loss: A Comprehensive Guide. Available at: https://www.v7labs.com/blog/cross-entropy-loss-guide (Accessed: 2024).

Yuan, Q., Yin, L., He, J., Zeng, Q., Liang, Y., Shen, Y. and Zu, X. (2024) Metabolism of asparagine in the physiological state and cancer. Cell Communication and Signaling, 22, p.163.

Zhang, Y., Hu, Y., Jiang, N. and Yetisen, A.K. (2023) Wearable artificial intelligence biosensor networks. Biosensors and Bioelectronics, 219, p.114825.